CLAIMS

1. An integrated circuit (IC) card comprising:

an input/output (I/O) interface; and

a smart card development interface, coupled to the I/O interface, to receive and identify one or more debug frames interlaced within a normal communication flow between the IC card and a host system.

2. An IC card according to claim 1, further comprising:

a memory device having stored therein a plurality of executable instructions; and

a controller, coupled to the memory device and the smart card development interface, to execute at least a subset of the plurality of executable instruction to selectively implement one or more of a plurality of IC card applets.

3. An IC card according to claim 2, wherein the memory device includes a plurality of executable instructions which, when executed, implement a debug application which selectively controls other applications executing on the IC card.

4. An IC card according to claim 1, wherein the smart card development interface includes a debug filter to identify and remove the debug frames from the normal communication flow.

5. An IC card according to claim 4, wherein the debug filter redirects the debug frames to a debug application on the IC card.

6. An IC card according to claim 1, further comprising a debug application, responsive to debug instructions embedded within received debug frames, the debug application providing a user with a host of application debug features enabled in response to the received debug instructions.

7. An IC card according to claim 6, wherein select debug instructions invoke one or more of the following debug features: read/write IC card memory, get/set breakpoints in an IC card applet, sequentially step an IC card application, run an IC card applet, and release an IC card applet frame.

8. An IC card according to claim 1, wherein the IC card communicates with a remote host system using a transport protocol comprising application data units (APDU) and debug protocol data units (DPDU).

9. An IC card according to claim 8, wherein the transport protocol is a standard smart card communication protocol, wherein the APDU and the DPDU adhere to the standard IC card communication protocol.

10. An IC card according to claim 8, wherein the smart card development interface further comprises a debug filter which identifies DPDU within the normal communication flow to redirect the DPDU to a debug application on the IC card.

11. An IC card according to claim 10, wherein the debug filter identifies DPDU within the normal communication flow by detecting an invalid source and/or destination address identifier within the debug frame.

12. An IC card according to claim 1, wherein the IC card communicates with a remote host system using a transport protocol comprising application data units (APDU) including normal application frames and debug frames.

13. A storage medium having stored thereon a plurality of executable instructions which, when executed, implement the smart card development interface of claim 1.

14. A method of debugging a smart card application, the method comprising:

receiving one or more debug frames interlaced with application frames comprising a normal communication flow between a smart card and a host system;

identifying the one or more debug frames;

routing the received debug frames to a debug application executing on the smart card, while promoting the application frames to an application executing on the smart card, subject to conditions imposed by the debug frames.

15. A method according to claim 14, wherein the step of identifying the one or more debug frames comprises:

reading a source and/or destination address of frames comprising the normal communication flow; and

detecting invalid source and/or destination addresses in select frames denoting debug frames.

16. A method according to claim 14, further comprising:

implementing one or more debug features on the smart card according to debug instructions embedded within the received debug frames.

17. A method according to claim 16, wherein the debug features include one or more of read/write smart card memory, get/set breakpoints in a smart card application, sequentially step a smart card application, run a smart card application, and release a smart card application frame.

18. A method according to claim 14, further comprising:

generating a response debug frame to a received debug frame;

interlacing the response debug frame with response application frames; and

sending the response debug frame and response application frames to a host system.

19. A storage medium having stored thereon a plurality of instructions which, when executed, implement the method of claim 14.

20. A computer system comprising:

an input/output (I/O) interface; and

a client development interface, coupled to the I/O interface, to receive and identify debug frames interlaced within the normal communication flow between the computer system and a removably coupled smart card.

21. A computer system according to claim 20, further comprising:

a memory device having stored therein a plurality of instructions; and

a processor, coupled to the memory device and the client development interface, to execute at least a subset of the plurality of instructions to implement one or more applications including a smart card development application having a debug environment to send and receive debug frames to the coupled IC card interlaced within the normal communication flow between the computer system and the IC card.

22. A computer system according to claim 14, wherein the memory device includes a plurality of executable instructions which, when executed, implement a debug application on the computer system to communicate with and control smart card resources.

23. A computer system according to claim 20, wherein the client development interface includes a debug filter to identify and remove the debug frames from the normal communication flow between the computer system and the smart card.

24. A computer system according to claim 23, wherein the debug filter redirects debug frames received from the smart card to a debug application executing on the computer system.

25. A computer system according to claim 20, further comprising a debug application, to write and read debug frames to and from the smart card, facilitating a number of application debugging features.

26. A computer system according to claim 25, wherein the debug frames written by the debug application invoke one or more of the following debug features: read/write smart card memory, get/set breakpoints in a smart card application, sequentially step a smart card application, run a smart card application, and release a smart card application frame.

27. A computer system according to claim 20, wherein the computer system communicates with the smart card using a transport protocol comprising application data units (APDU) and debug protocol data units (DPDU).

28. A computer system according to claim 27, wherein the client development interface further comprises a debug filter which identifies DPDU within the normal communication flow to redirect the DPDU to a debug application executing on the computer system.

29. A computer system according to claim 29, wherein the debug filter identifies DPDU within the normal communication flow by detecting an invalid source and/or destination address identifier within the debug frame.

30. A storage medium having stored thereon a plurality of executable instructions which, when executed, implement the client development interface of claim 20.

31. A computer-implemented method for debugging a smart card application, the method comprising:

generating one or more debug frames containing debug instructions;

interlacing the generated debug frames with one or more application frames generated according to an application executing on the computer; and

sending the application frames with the interlaced debug frames to a removably coupled smart card, wherein the debug frames invoke one or more debug features of the smart card.

32. A computer-implemented method according to claim 31, wherein the application frames are generated by an application executing within an application development environment, while the debug frames are generated in response to user interaction with the smart card application development environment.

33. A computer-implemented method according to claim 31, wherein generating one or more debug frames comprises populating a source and/or destination field of the debug frame with an invalid source and/or destination address.

34. A computer-implemented method according to claim 31, further comprising:

receiving a normal communication flow from the smart card including debug frames interlaced with application frames, wherein the debug frames received from the smart card are received in response to debug frames issued by the computer.

35. A computer-implemented method according to claim 34, wherein the application frames are promoted to an associated application executing within an application development tool executing on the computer, while the debug frames are promoted to an application debug environment of the application development tool executing on the computer.

36. A communication protocol, employed between a host system and a smart card, the protocol comprising:

a plurality of application frames comprising a normal communication flow between a host application and a smart card application; and

one or more debug frames, interlaced with the application frames within the normal communication flow, to enable a debug application executing on the host system to selectively access and control smart card resources.

37. A communication protocol according to claim 36, wherein the debug application and the host application are executing on separate host systems, each communicatively coupled to the smart card.

38. A communication protocol according to claim 36, wherein the one or more debug frames include debug instruction to implement one or more of the following debug features: read/write smart card memory, get/set breakpoints in a smart card application, sequentially step a smart card application, run a smart card application, and release a smart card application frame.

39. A communication protocol according to claim 36, wherein the debug frame is distinguished from an application frame by incorporating an invalid source address.

40. A communication protocol according to claim 36, wherein the debug frame is distinguished from an application frame by incorporating an invalid destination address.

41. An application development system comprising:

a computer system to execute an application within an application development tool; and

a smart card incorporating a smart card development interface, coupled to the computer system, to receive and identify debug frames interlaced with application frames within a normal communication flow between the application executing on the computer system and the smart card, wherein the smart card development interface promotes the application frames to an application layer of the smart card, and invokes debug features of the smart card in response to debug instructions embedded within the received debug frames.

42. An application development system according to claim 41, wherein the computer system further comprises:

a client development interface, to interlace debug frames generated by the application development tool with application frames generated by the application executing within the application development tool.

43. An application development system according to claim 42, wherein the application development tool generates debug frames in response to user interaction with the application development tool.

44. An application development system according to claim 43, wherein the application development tool populates a source and/or destination field of the debug frame with an invalid source and/or destination address.

45. An application development system according to claim 43, wherein the debug frames invoke and control one or more smart card resources facilitating debugging of the application executing within the application development tool of the computer system.

46. An application development system according to claim 42, wherein the client development interface includes a debug filter to identify and route debug frames received from the smart card.

47. An application development system according to claim 41, wherein the smart card development interface comprises a debug filter to identify debug frames within the received normal communication flow.

48. An application development system according to claim 47, wherein the debug filter identifies debug frames by an invalid source and/or destination address embedded within a source and/or destination field of the debug frame.

49. An application development system according to claim 41, further comprising:

a communication protocol, employed by the computer system and the smart card to communicate therebetween, the communication protocol comprising,

a plurality of application frames comprising a normal communication flow between a host application and a smart card application; and

one or more debug frames, interlaced with the application frames within the normal communication flow, to enable a debug application executing on the host system to selectively access and control smart card resources.